

Centro Universitário UNISATC

Engenharia de Software 3a fase – Banco de Dados II – Prof. Jorge Luiz da Silva

TRABALHO FINAL COM BASE EM METODOLOGIAS ATIVAS DE APRENDIZAGEM

Projeto de banco de dados para um sistema de segurança para celulares

Yasmin Bez Fontana - @bfontanayasmin

Felipe Saturno - @f2004felipe

Nicolas Pereira - @hqnicolas

Cleiton Lino Rodrigues - @TonClei91



Axel Filastro - @Axelfilastro

Evandro Luiz Rodrigues Damazio - @evandroluiz


Criciúma, 03/12/2024




URL do projeto no GitHub


https://github.com/hqnicolas/bd2


 hqnicolas / bd2


[Code](#) [Issues 5](#) [Pull requests](#) [Actions](#) [Projects](#) [Wiki](#) [Security](#) [Insights](#)



 **bd2** Public Watch 2









 main  1 Branch  0 Tags



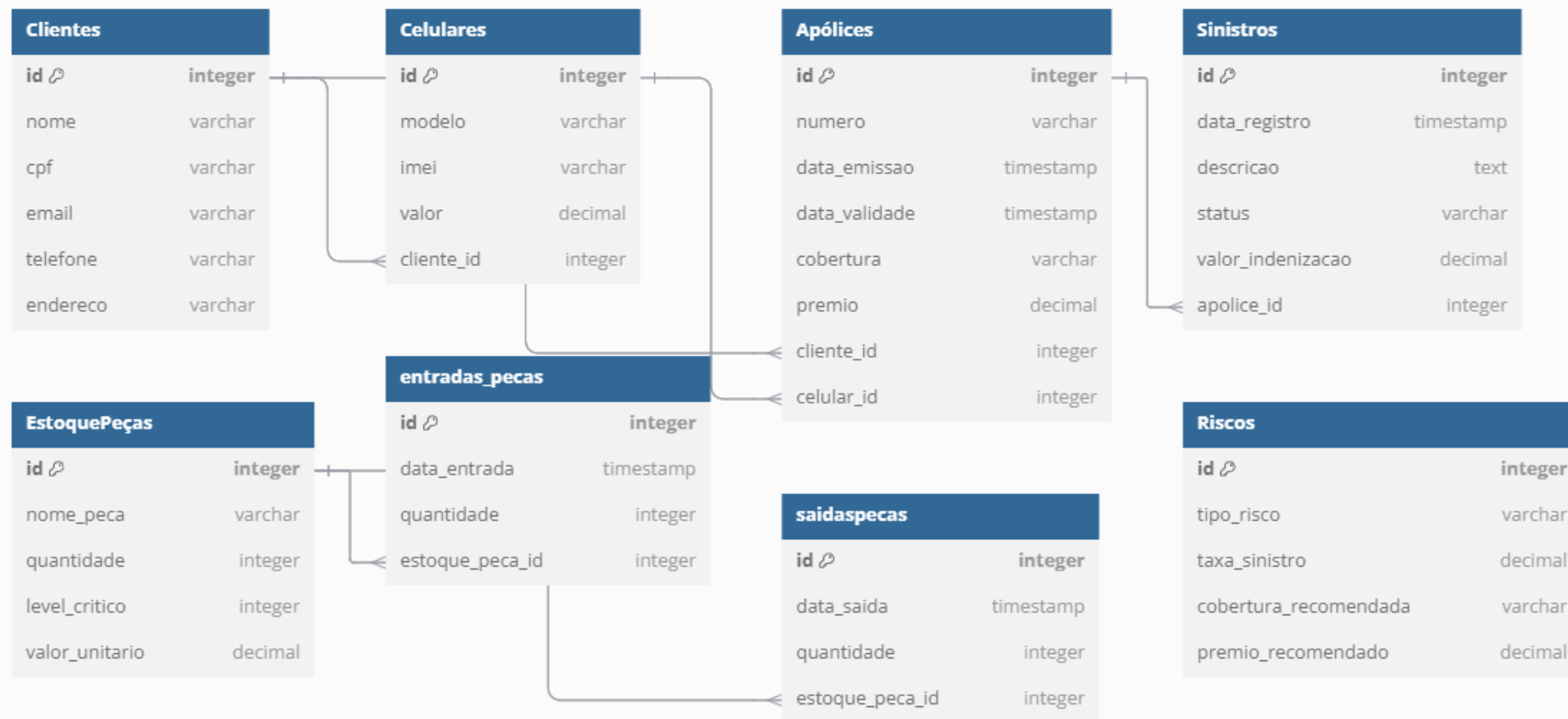
[Add file](#) 

[Code](#) 

 **bfontanayasmin** Update dml.sql b65876c · 12 minutes ago  39 Commits

 docs	Dicionário de dados atualizado	32 minutes ago
 public	fix: promote integration and navigation between front-end s...	1 hour ago
 src	Update dml.sql	12 minutes ago
 Dockerfile	backend inside docker #8	2 weeks ago
 README.md	synk project files	5 days ago
 app.js	feat: building the backend with individual scripts from each ...	53 minutes ago
 docker-compose.yml	backend inside docker #8	2 weeks ago
 package.json	backend inside docker #8	2 weeks ago

Modelo ER Físico



Dicionário de Dados

Tabela	clients					
Descrição	Tabela responsável por armazenar os dados das avaliações criadas pelos professores					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação do cliente.
nome	varchar(255)	1-255	NOT NULL			Nome do cliente
cpf	varchar(255)	1-255	NOT NULL			CPF do cliente
email	varchar(255)	1-255	NOT NULL			E-mail do cliente
telefone	varchar(255)	1-255	NULL			Telefone do cliente
endereço	varchar(255)	1-255	NULL			Endereço do cliente
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_clients_email		X		email		
Tabela	celulares					
Descrição	Tabela responsável por armazenar os dados dos celulares inseridos.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação do Celular.
modelo	varchar(255)	1-255	NOT NULL			Modelo do celular.
imei	varchar(255)	1-255	NOT NULL			IMEI do celular.
valor	decimal(10,2)	0.00 - 100000.00	NOT NULL			Valor do celular.
cliente_id	integer	1 a 2147483647	NOT NULL		X	Cliente pertencente.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_imei		X		imei		
idx_cliente_id		X		cliente_id		
Tabela	apolices					

Descrição	Tabela responsável por armazenar os dados das apólices.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação do Apólice.
numero	varchar(100)	1-100	NOT NULL			Número da apólice
data_emissao	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Data de emissão da apólice.
data_validade	decimal(10,2)	0.00 - 100000.00	NOT NULL			Data de validade da apólice.
cobertura	varchar(255)	1-255	NOT NULL			Cobertura da apólice.
premio	decimal(10,2)	0.00 - 100000.00	NOT NULL			Valor do premio.
cliente_id	integer	1-2147483647	NOT NULL		X	Cliente pertencente.
celular_id	integer	1-2147483647	NOT NULL		X	Celular pertencente.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_apolices_numero		X		numero		
idx_apolices_cliente_id		X		cliente_id		
idx_apolices_celular_id		X		celular_id		
Tabela	sinistros					
Descrição	Tabela responsável por armazenar os dados dos sinistros.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação do sinistro.
data_registro	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Data do registro do sinistro.
descricao	text	1-65.535	NOT NULL			Descrição do sinistro.
status	varchar(50)	1-50.	NOT NULL			Status do sinistro.
valor_indenizacao	decimal(10,2)	0.00 - 100000.00	NULL			Valor de indenização do sinistro.
apolice_id	integer	1-2147483647	NOT NULL		X	Apólice pertencente.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_apolice_id		X		apolice_id		

Tabela	estoque_pecas					
Descrição	Tabela responsável por armazenar o estoque de peças.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação da peça.
nome_peca	varchar(255)	1-255	NOT NULL			Nome da peça.
quantidade	integer	1-2147483647	NOT NULL			Quantidade disponível da peça
level_critico	integer	1-2147483647	NOT NULL			Quantidade do nível crítico.
valor_unitario	decimal(10,2)	0.00 - 100000.00	NOT NULL			Valor unitário da peça.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.
Tabela	entrada_pecas					
Descrição	Tabela responsável por armazenar os dados das entradas de peças do estoque.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação da entrada.
data_entrada	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Data da entrada.
quantidade	integer	1-2147483647	NOT NULL			Quantidade entrada.
estoque_peca_id	integer	1-2147483647	NOT NULL		X	Estoque pertencente.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.
Índice						
Nome do índice	Clustered	NonClustered	Unique	Colunas		
idx_entradas_pecas_estoque_peca_id		X		estoque_peca_id		
Tabela	saida_pecas					
Descrição	Tabela responsável por armazenar os dados das saídas de peças do estoque.					
Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação da saída.
data_saida	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Data da saída.
quantidade	integer	1-2147483647	NOT NULL			Quantidade que saiu.
estoque_pecas_id	integer	1-2147483647	NOT NULL		X	Estoque pertencente.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.

Índice				
Nome do índice	Clustered	NonClustered	Unique	Colunas
idx_saidas_pecas_estoque_peca_id		X		estoque_peca_id

Tabela	riscos
Descrição	Tabela responsável por armazenar os dados dos riscos.

Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação do risco.
tipo_risco	varchar(255)	1-255	NOT NULL			Tipo do risco.
taxa_sinistro	decimal(10,2)	0.00 - 100000.00	NOT NULL			Taxa do sinistro.
cobertura_recomendada	varchar(255)	1-255	NOT NULL			Cobertura recomendada.
premio_recomendado	decimal(10,2)	0.00 - 100000.00	NOT NULL			Valor do prêmio recomendado.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.

Tabela	ApoliceLog
Descrição	Tabela responsável por armazenar os dados deletados.

Atributos						
Nome da Coluna	Tipo do Dado	Valor min e max	Nulidade	PK	FK	Descrição
id	serial	1 a 2147483647	NOT NULL	X		Código de identificação.
numero	varchar(50)	1-50.	NOT NULL			Número de identificação.
data_emissao	date	1000-01-01 a 9999-12-31	NOT NULL			Data de emissão da apólice.
data_validade	date	1000-01-01 a 9999-12-31	NOT NULL			Data de validade da apólice.
cobertura	varchar(100)	0-100	NOT NULL			Cobertura recomendada.
premio	decimal(10,2)	0.00 - 100000.00	NOT NULL			Prêmio recomendado.
cliente_id	integer	1000-01-01 a 9999-12-31	NOT NULL		X	Cliente pertencente.
celular_id	date	1000-01-01 a 9999-12-31	NOT NULL		X	Celular pertencente.
deleted_date	timestamp	0-100	NOT NULL			Data que foi deletada.
deleted_by	decimal(10,2)	0.00 - 100000.00	NOT NULL			Quem deletou.
created_at	timestamp	1000-01-01 00:00:00 a 9999-12-31 23:59:59	NOT NULL			Horário da criação.

Script dos comandos DDL para criação do Banco de dados

-- Tabela: clients

CREATE TABLE IF NOT EXISTS clients (id SERIAL **PRIMARY KEY**, nome **VARCHAR(255) NOT NULL**, cpf **VARCHAR(255) UNIQUE NOT NULL**, email **VARCHAR(255) UNIQUE NOT NULL**, telefone **VARCHAR(255)**, endereco **VARCHAR(255)**, created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);-- Tabela: celulares

CREATE TABLE IF NOT EXISTS celulares (id SERIAL **PRIMARY KEY**, modelo **VARCHAR(255) NOT NULL**, imei **VARCHAR(255) UNIQUE NOT NULL**, valor **DECIMAL(10, 2) NOT NULL**, cliente_id **INTEGER REFERENCES** clients(id), created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: apolices

CREATE TABLE IF NOT EXISTS apolices (id SERIAL **PRIMARY KEY**, numero **VARCHAR(100) UNIQUE NOT NULL**, data_emissao **TIMESTAMP NOT NULL**, data_validade **TIMESTAMP NOT NULL**, cobertura **VARCHAR(255) NOT NULL**, premio **DECIMAL(10, 2) NOT NULL**, cliente_id **INTEGER REFERENCES** clients(id), celular_id **INTEGER REFERENCES** celulares(id), created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: sinistros

CREATE TABLE IF NOT EXISTS sinistros (id SERIAL **PRIMARY KEY**, data_registro **TIMESTAMP NOT NULL**, descricao **TEXT NOT NULL**, status **VARCHAR(50) NOT NULL**, valor_indenizacao **DECIMAL(10, 2)**, apolice_id **INTEGER REFERENCES** apolices(id), created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: estoque_pecas

CREATE TABLE IF NOT EXISTS estoque_pecas (id SERIAL **PRIMARY KEY**, nome_peca **VARCHAR(255) NOT NULL**, quantidade **INTEGER NOT NULL**, level_critico **INTEGER NOT NULL**, valor_unitario **DECIMAL(10, 2) NOT NULL**, created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: entradas_pecas

CREATE TABLE IF NOT EXISTS entradas_pecas (id SERIAL **PRIMARY KEY**, data_entrada **TIMESTAMP NOT NULL**, quantidade **INTEGER NOT NULL**, estoque_peca_id **INTEGER REFERENCES** estoque_pecas(id), created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: saidas_pecas

CREATE TABLE IF NOT EXISTS saidas_pecas (id SERIAL **PRIMARY KEY**, data_saida **TIMESTAMP NOT NULL**, quantidade **INTEGER NOT NULL**, estoque_peca_id **INTEGER REFERENCES** estoque_pecas(id), created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: riscos

CREATE TABLE IF NOT EXISTS riscos (id SERIAL **PRIMARY KEY**, tipo_risco **VARCHAR(255) NOT NULL**, taxa_sinistro **DECIMAL(10, 2) NOT NULL**, cobertura_recomendada **VARCHAR(255) NOT NULL**, premio_recomendado **DECIMAL(10, 2) NOT NULL**, created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: ApoliceLog

CREATE TABLE IF NOT EXISTS ApoliceLog (id SERIAL **PRIMARY KEY**, numero **VARCHAR**(50) **NOT NULL**, data_emissao **DATE NOT NULL**, data_validade **DATE NOT NULL**, cobertura **VARCHAR**(100) **NOT NULL**, premio **DECIMAL**(10, 2) **NOT NULL**, cliente_id **INTEGER NOT NULL**, celular_id **INTEGER NOT NULL**, deleted_date **TIMESTAMP NOT NULL**, deleted_by **VARCHAR**(50) **NOT NULL**, created_at **TIMESTAMP DEFAULT CURRENT_TIMESTAMP**);

-- Tabela: metadata

CREATE TABLE IF NOT EXISTS metadata (initialized **BOOLEAN DEFAULT FALSE**);

Script que popula as tabelas do Banco de dados

-- Instruções DML para gerenciar registros de banco de dados

-- Inserindo dados iniciais dos clientes

```
INSERT INTO clients (nome, cpf, email, telefone, endereco) VALUES ('Felipe Castro', '123.456.789-01', 'felipe.castro@hotmail.com', '(11) 9.1234-5678', 'Rua Exemplo, 123'), ('Rafaela Vieira', '987.654.321-09', 'rafaela.vieira@hotmail.com', '(11) 8.9012-3456', 'Avenida Teste, 456'), ('João Silva', '741.852.863-27', 'joao.silva@hotmail.com', '(11) 7.4567-8901', 'Rua Amostra, 789'), ('Maria Luiza', '369.258.147-85', 'maria.luiza@hotmail.com', '(11) 6.7890-1234', 'Rua Modelo, 101'), ('Pedro Henrique', '852.913.741-52', 'pedro.henrique@hotmail.com', '(11) 5.6789-0456', 'Avenida Padrão, 202'), ('Ana Clara', '456.789.012-44', 'ana.clara@hotmail.com', '(11) 4.5678-9012', 'Rua Referência, 303'), ('Lucas Mateus', '321.754.987-65', 'lucas.mateus@hotmail.com', '(11) 3.4567-8901', 'Rua Exemplos, 404'), ('Beatriz Farias', '147.252.963-74', 'beatriz.farias@hotmail.com', '(11) 2.7890-1234', 'Avenida Demonstração, 505'), ('Gabriel Oliveira', '863.741.852-93', 'gabriel.oliveira@hotmail.com', '(11) 1.6789-0456', 'Rua Ilustração, 606'), ('Juliana Alves', '741.842.863-27', 'juliana.alves@hotmail.com', '(11) 9.8765-4321', 'Rua Amostragem, 707'), ('Rafael Souza', '852.923.741-52', 'rafael.souza@hotmail.com', '(11) 8.7654-3210', 'Avenida Padrão, 808'), ('Caroline Dias', '456.789.412-34', 'caroline.dias@hotmail.com', '(11) 7.6543-2109', 'Rua Referência, 909'), ('Eduardo Santos', '321.654.985-65', 'eduardo.santos@hotmail.com', '(11) 6.5432-1098', 'Rua Exemplos, 1010'), ('Larissa Ferreira', '147.852.953-74', 'larissa.ferreira@hotmail.com', '(11) 5.4321-0987', 'Avenida Demonstração, 1111'), ('Vinicius Costa', '963.742.852-93', 'vinicius.costa@hotmail.com', '(11) 4.3210-9876', 'Rua Ilustração, 1212'), ('Juliana Silva', '741.852.963-27', 'juliana.silva@hotmail.com', '(11) 3.2109-8765', 'Rua Amostragem, 1313'), ('Leonardo Almeida', '852.963.744-52', 'leonardo.almeida@hotmail.com', '(11) 2.1098-7654', 'Avenida Padrão, 1414'), ('Isabella Rodrigues', '456.489.012-34', 'isabella.rodrigues@hotmail.com', '(11) 1.0987-6543', 'Rua Referência, 1515'), ('Daniel Martins', '322.654.987-65', 'daniel.martins@hotmail.com', '(11) 9.9876-5432', 'Rua Exemplos, 1616'), ('Giovanna Lima', '147.853.963-74', 'giovanna.lima@hotmail.com', '(11) 8.8765-4321', 'Avenida Demonstração, 1717'), ('Mateus Sousa', '964.741.852-93', 'mateus.sousa@hotmail.com', '(11) 7.7654-3210', 'Rua Ilustração, 1818');
```

-- Inserindo dados iniciais dos celulares

```
INSERT INTO celulares (modelo, imei, valor, cliente_id, created_at) VALUES ('Samsung S21', '1234567898123456', 1500.00, 1, NOW()), ('Apple iPhone 13', '9876543210887654', 2000.00, 1, NOW()), ('Xiaomi Redmi 9', '5432103876543210', 300.00, 2, NOW()), ('Google Pixel 6', '9876543210487654', 1800.00, 3, NOW()), ('Huawei P30', '5435109876543210', 1200.00, 4, NOW()), ('OnePlus 9 Pro', '9874543210987654', 2500.00, 5, NOW()), ('Oppo Reno Ace', '5432109876543610', 800.00, 6, NOW()), ('Vivo X50 Pro', '9876943210987654', 1000.00, 7, NOW()), ('Motorola G Power', '5432105876543210', 400.00, 8, NOW()), ('Nokia 3310', '9876541210987654', 100.00, 9, NOW()), ('LG G8X ThinQ', '5432101876543210', 1500.00, 10, NOW()), ('Sony Xperia 1 III', '9876513210987654', 2000.00, 11, NOW()), ('Asus Zenfone 8', '5432109876543210', 800.00, 12, NOW()), ('Realme GT Master', '9826543210987654', 600.00, 13, NOW()), ('TCL 10L', '5431109876543210', 300.00, 14, NOW()), ('ZTE Axon 30 Ultra', '9876543210987854', 1200.00, 15, NOW()), ('Honor Magic4 Ultimate', '5432109876243210', 1800.00, 16, NOW()), ('Xiaomi Mi 11 Ultra', '9876343210987654', 2500.00, 17, NOW()), ('Oppo Find X4 Pro', '5532109876543210', 2000.00, 18, NOW());
```

-- Inserindo dados iniciais do estoque de peças

```
INSERT INTO estoque_pecas (nome_pecas, quantidade, level_critico, valor_unitario, created_at) VALUES ('Lâmpada', 100, 5, 5.00, NOW()), ('Filtro de Ar', 50, 3, 10.00, NOW()), ('Bateria', 200, 4, 8.00, NOW()), ('Pneu', 150, 2, 20.00, NOW()), ('Kit de Ferramentas', 30, 1, 50.00, NOW()), ('Óleo Motor', 120, 6, 15.00, NOW()), ('Filtro de Combustível', 40, 7, 12.00, NOW()), ('Câmera de Segurança', 80, 8, 30.00, NOW()), ('Sensor de Movimento', 60, 9, 25.00, NOW());
```

-- Inserindo dados iniciais das entradas de peças

```
INSERT INTO entradas_pecas (data_entrada, quantidade, estoque_pecas_id, created_at) VALUES ('2023-01-01', 20, 1, NOW()), ('2023-01-15', 10, 2, NOW()), ('2023-02-01', 15, 3, NOW()), ('2023-02-15', 12, 4, NOW()), ('2023-03-01', 18, 5, NOW()), ('2023-03-15', 14, 6, NOW()), ('2023-04-01', 16, 7, NOW()), ('2023-04-15', 11, 8, NOW());
```

-- Inserindo dados iniciais dos apolices

```
INSERT INTO apolices (numero, data_emissao, data_validade, cobertura, premio, cliente_id, celular_id, created_at) VALUES ('APOLICE-001', '2023-01-01', '2024-01-01', 'Seguro de todas as peças', 1000.00, 1, 1, NOW()), ('APOLICE-002', '2023-02-01', '2024-02-01', 'Seguro de bateria', 500.00, 2, 2, NOW()), ('APOLICE-003', '2023-03-01', '2024-03-01', 'Seguro de câmera', 800.00, 3, 3, NOW()), ('APOLICE-004', '2023-04-01', '2024-04-01', 'Seguro de roubo', 600.00, 4, 4, NOW()), ('APOLICE-005', '2023-05-01', '2024-05-01', 'Seguro de todas as peças', 1000.00, 5, 5, NOW()), ('APOLICE-006', '2023-06-01', '2024-06-01', 'Seguro de carcaça', 500.00, 6, 6, NOW()), ('APOLICE-007', '2023-07-01', '2024-07-01', 'Seguro de câmera', 800.00, 7, 7, NOW()), ('APOLICE-008', '2023-08-01', '2024-08-01', 'Seguro de roubo', 600.00, 8, 8, NOW()), ('APOLICE-009', '2023-09-01', '2024-09-01', 'Seguro de todas as peças', 1000.00, 9, 9, NOW()), ('APOLICE-010', '2023-10-01', '2024-10-01', 'Seguro de carcaça', 500.00, 10, 10, NOW()), ('APOLICE-011', '2023-11-01', '2024-11-01', 'Seguro de câmera', 800.00, 11, 11, NOW()), ('APOLICE-012', '2023-12-01', '2024-12-01', 'Seguro de roubo', 600.00, 12, 12, NOW());
```

-- Inserindo dados iniciais das saídas de peças

```
INSERT INTO saidas_pecas (data_saida, quantidade, estoque_pecas_id, created_at) VALUES ('2023-01-01', 5, 1, NOW()), ('2023-01-15', 3, 2, NOW()), ('2023-02-01', 4, 3, NOW()), ('2023-02-15', 2, 4, NOW()), ('2023-03-01', 6, 5, NOW()), ('2023-03-15', 5, 6, NOW()), ('2023-04-01', 7, 7, NOW()), ('2023-04-15', 4, 8, NOW());
```

-- Inserindo dados iniciais dos sinistros

```
INSERT INTO sinistros (data_registro, descricao, status, valor_indenizacao, apolice_id, created_at) VALUES ('2023-01-15', 'Quebra do vidro do carro', 'Aberto', 200.00, 1, NOW()), ('2023-02-20', 'Doença grave', 'Encerrado', 1000.00, 2, NOW()), ('2023-03-25', 'Incêndio em casa', 'Encerrado', 800.00, 3, NOW()), ('2023-04-30', 'Acidente de trânsito', 'Aberto', 500.00, 4, NOW()), ('2023-05-05', 'Morte de
```

```
um familiar', 'Encerrado', 1000.00, 5, NOW()), ('2023-06-10', 'Furto de objetos pessoais', 'Aberto', 200.00, 6, NOW()), ('2023-07-15', 'Lesão corporal', 'Encerrado', 600.00, 7, NOW()), ('2023-08-20', 'Dano ao imóvel', 'Aberto', 800.00, 8, NOW()), ('2023-09-25', 'Doença grave', 'Encerrado', 1000.00, 9, NOW()), ('2023-10-30', 'Acidente de trânsito', 'Aberto', 500.00, 10, NOW()), ('2023-11-05', 'Morte de um familiar', 'Encerrado', 1000.00, 11, NOW()), ('2023-12-10', 'Furto de objetos pessoais', 'Aberto', 200.00, 12, NOW());
```

```
-- Inserindo dados iniciais dos riscos
```

```
INSERT INTO riscos (tipo_risco, taxa_sinistro, cobertura_recomendada, premio_recomendado, created_at) VALUES ('Fogo', 0.05, 'Seguro de câmara', 800.00, NOW()), ('Inundação', 0.03, 'Seguro de câmara', 600.00, NOW()), ('Queda', 0.02, 'Seguro de carcaça', 500.00, NOW()), ('tela', 0.01, 'Seguro de roubo', 1000.00, NOW()), ('Queda', 0.005, 'Seguro de todas as peças', 1000.00, NOW());
```

```
INSERT INTO clients (nome, cpf, email, telefone, endereco) VALUES
```

```
('Carlos Eduardo', '987.654.321-11', 'carlos.eduardo@hotmail.com', '(11) 9.8765-4321', 'Rua Nova, 212'),  
( 'Fernanda Lima', '321.987.654-22', 'fernanda.lima@hotmail.com', '(11) 8.7654-3210', 'Avenida Principal, 314'),  
( 'André Santos', '654.321.987-33', 'andre.santos@hotmail.com', '(11) 7.6543-2109', 'Rua Secundária, 415'),  
( 'Patrícia Souza', '987.654.123-44', 'patricia.souza@hotmail.com', '(11) 6.5432-1098', 'Rua Central, 516'),  
( 'Lucas Almeida', '321.456.789-55', 'lucas.almeida@hotmail.com', '(11) 5.4321-0987', 'Avenida Paralela, 617');
```

```
INSERT INTO celulares (modelo, imei, valor, cliente_id, created_at) VALUES
```

```
('Samsung Galaxy A72', '1234569876543210', 1500.00, 20, NOW()),  
( 'Apple iPhone SE', '9876543210981123', 2500.00, 21, NOW()),  
( 'Xiaomi Poco X3', '6543210987654321', 1200.00, 22, NOW()),  
( 'OnePlus Nord 2', '1234987654321098', 2200.00, 23, NOW()),  
( 'Motorola Edge 20', '9876123456784321', 1800.00, 24, NOW());
```

```
INSERT INTO estoque_pecas (nome_peca, quantidade, level_critico, valor_unitario, created_at) VALUES
```

```
('Velas de Ignição', 70, 10, 15.00, NOW()),  
( 'Fusível', 300, 20, 2.00, NOW()),
```

```
('Radiador', 40, 5, 150.00, NOW()),  
( 'Filtro de Óleo', 90, 8, 25.00, NOW()),  
( 'Correia Dentada', 60, 10, 100.00, NOW());
```

```
INSERT INTO entradas_pecas (data_entrada, quantidade, estoque_peca_id, created_at) VALUES  
( '2024-01-01', 25, 10, NOW()),  
( '2024-01-15', 15, 11, NOW()),  
( '2024-02-01', 10, 12, NOW()),  
( '2024-02-15', 18, 13, NOW()),  
( '2024-03-01', 20, 14, NOW());
```

```
INSERT INTO apolices (numero, data_emissao, data_validade, cobertura, premio, cliente_id, celular_id, created_at) VALUES  
( 'APOLICE-013', '2023-07-01', '2024-07-01', 'Seguro de tela', 600.00, 20, 20, NOW()),  
( 'APOLICE-014', '2023-08-01', '2024-08-01', 'Seguro de bateria', 400.00, 21, 21, NOW()),  
( 'APOLICE-015', '2023-09-01', '2024-09-01', 'Seguro total', 1200.00, 22, 22, NOW()),  
( 'APOLICE-016', '2023-10-01', '2024-10-01', 'Seguro contra roubo', 800.00, 23, 23, NOW()),  
( 'APOLICE-017', '2023-11-01', '2024-11-01', 'Seguro contra danos líquidos', 900.00, 24, 24, NOW());
```

```
INSERT INTO saidas_pecas (data_saida, quantidade, estoque_peca_id, created_at) VALUES  
( '2024-01-01', 10, 10, NOW()),  
( '2024-01-15', 5, 11, NOW()),  
( '2024-02-01', 7, 12, NOW()),  
( '2024-02-15', 8, 13, NOW()),  
( '2024-03-01', 6, 14, NOW());
```

```
INSERT INTO sinistros (data_registro, descricao, status, valor_indenizacao, apolice_id, created_at) VALUES
('2023-07-15', 'Quebra da tela', 'Aberto', 400.00, 13, NOW()),
('2023-08-20', 'Problema na bateria', 'Encerrado', 300.00, 14, NOW()),
('2023-09-25', 'Roubo do dispositivo', 'Encerrado', 1000.00, 15, NOW()),
('2023-10-30', 'Dano por queda', 'Aberto', 700.00, 16, NOW()),
('2023-11-05', 'Dano por líquido', 'Encerrado', 800.00, 17, NOW());
```

Principais consultas mapeadas baseadas em regras de negócio (mínimo 4)

Quantas apólices foram excluídas no último semestre e quais os detalhes registrados no log de exclusões?

```
SELECT COUNT(*) AS total_excluidas, cliente_id, cobertura, premio, deleted_date, deleted_by FROM ApoliceLog WHERE deleted_date >= DATE_TRUNC('month', CURRENT_DATE) - INTERVAL '6 months' GROUP BY cliente_id, cobertura, premio, deleted_date, deleted_by ORDER BY deleted_date DESC;
```

Qual é o impacto financeiro acumulado de apólices deletadas, considerando as coberturas e prêmios associados?

```
SELECT SUM(premio) AS impacto_financeiro, cobertura, COUNT(*) AS total_apolices FROM ApoliceLog GROUP BY cobertura ORDER BY impacto_financeiro DESC;
```

O cliente teve algum sinistro registrado no ano anterior e qual foi o impacto financeiro associado?

```
SELECT c.nome AS cliente_nome, COUNT(s.id) AS total_sinistros, SUM(s.valor_indenizacao) AS impacto_financeiro FROM clients c LEFT JOIN apolices a ON c.id = a.cliente_id LEFT JOIN sinistros s ON a.id = s.apolice_id AND EXTRACT(YEAR FROM s.data_registro) = EXTRACT(YEAR FROM CURRENT_DATE) - 1 WHERE c.id = 20 -- Substitua pelo ID do cliente desejado GROUP BY c.nome;
```

Quais clientes possuem apólices expiradas ou que irão expirar no mês vigente?

```
SELECT c.nome AS cliente, c.email, c.telefone, a.numero AS apolice_numero, a.data_validade FROM clientes c JOIN apolices a ON c.id = a.cliente_id WHERE a.data_validade < CURRENT_DATE -- Apólices já expiradas OR (EXTRACT(YEAR FROM a.data_validade) = EXTRACT(YEAR FROM CURRENT_DATE) AND EXTRACT(MONTH FROM a.data_validade) = EXTRACT(MONTH FROM CURRENT_DATE)); -- Expiram no mês vigente
```